

EXHIBIT B

MSN	Exchange Always-Up-To- Date Notifications Delivery	Design Spec
Program Manager	Paul Limont (plimont)	
Developers	Srinivas Manda (smanda)	
Testers	Steve Mattox (smattox)	

1 Overview	2
1.1 Summary and Scope.....	2
1.1.1 Goals.....	2
1.1.2 Non-Goals.....	2
1.2 Dependencies and Partners.....	2
2 Design Overview.....	3
2.1 Introduction.....	3
2.2 Architecture.....	4
2.3 Interfaces.....	5
2.4 Implementation.....	5
2.5 Reliability	5
2.6 Extensibility	5
2.7 Configuration	6
3 Functional Blocks and Components.....	6
3.1 SMTP Store Driver Sink.....	6
3.1.1 Store Driver Sink Project.....	6
3.1.2 ISMTPStoreDriver Interface.....	6
3.1.3 IEventIsCacheable Interface.....	7
3.1.4 Local Delivery Processing	7
3.2 Delivery Listener	8
3.2.1 GetRequestType	8
3.2.2 HandleHttpRequest	8
3.2.3 ParseExOmaNotification	8
3.2.4 CExOmaParse.....	8
3.3 Delivery Routing Filter.....	8
4 Interfaces and Schemas.....	9
4.1 Notification Submission	9
4.1.1 AUTD Notification Schema.....	9
4.1.2 AUTD SMTP Message Format.....	10
4.2 Delivery Listener Interface.....	11
4.3 Delivery Routing Interface	12
4.4 IMobileMsg Submission	13
5 Security.....	13
5.1 Exchange-To-MSN Datacenter Interface	13
5.2 Delivery Protocols.....	13
6 Scalability and Topologies	13

7 Performance.....	13
8 Issues.....	13
9 Glossary	13
10 References	14
11 History.....	14

1 Overview

1.1 Summary and Scope

The document defines the design of MSN software that will receive Always-Up-To-Date (AUTD) information from corporate installations of Microsoft Exchange's Titanium release, repackage and deliver the information to partner carriers for transmission to carrier client PocketPC and Smartphone devices. This software will reside on MSN datacenter machines and handle traffic from corporate clients to carrier partners.

1.1.1 Goals

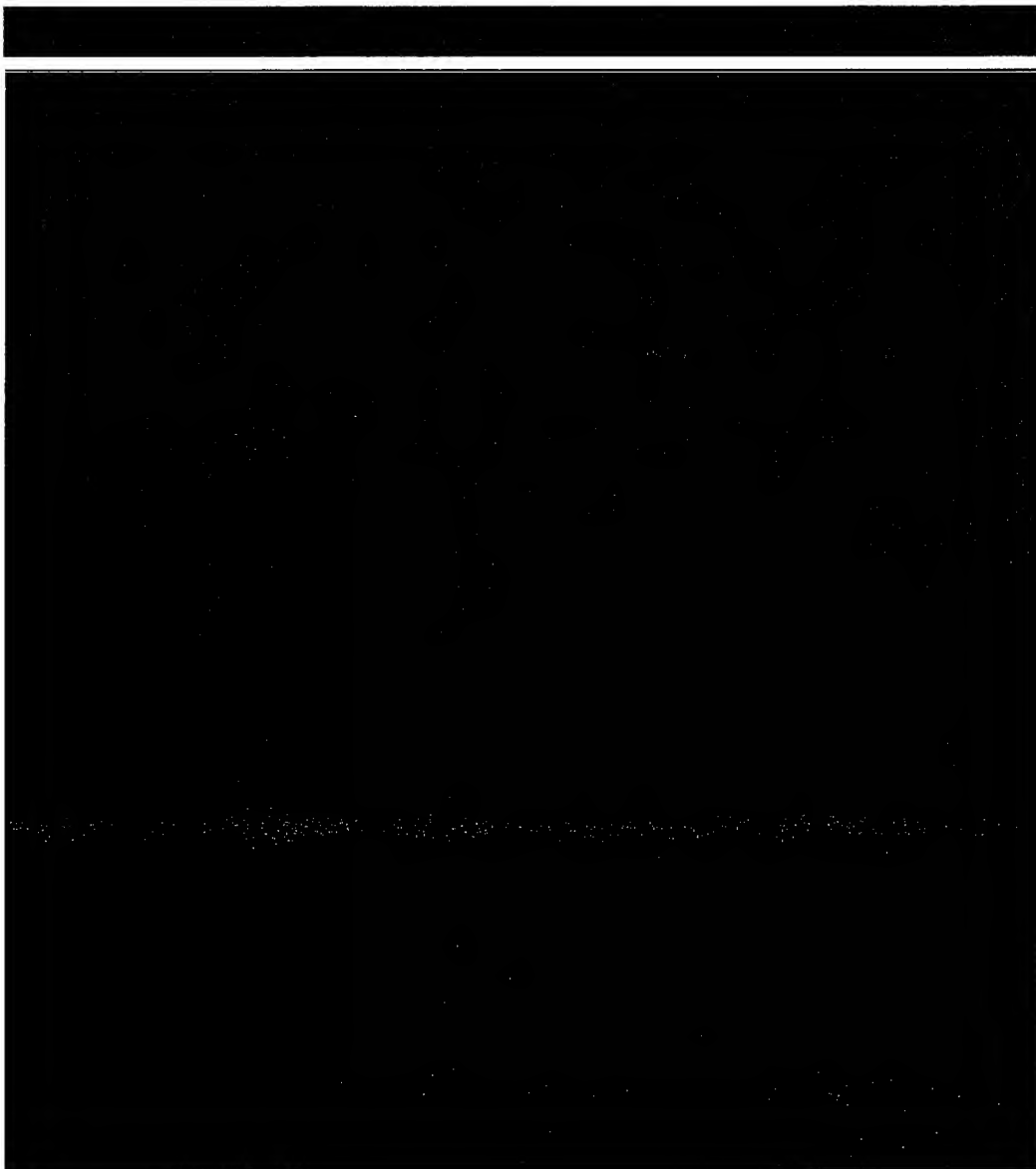
This document has the following goals:

- Define the interfaces associated with incoming AUTD information from corporate installations of Exchange Titanium
- Define the interfaces between internal MSN components that comprise the AUTD delivery path
- Define the interfaces associated with outgoing AUTD information to partner carriers
- Describe the design of internal MSN components that comprise the AUTD delivery path.

1.1.2 Non-Goals

This document specifically does not address the following issues:

- Design of any Exchange content delivery beyond that necessary for pinging PocketPC/Smartphone devices so that it can make a determination to initiate the synchronization process with an appropriate corporate Exchange server.
- Design of any software outside of the MSN delivery architecture

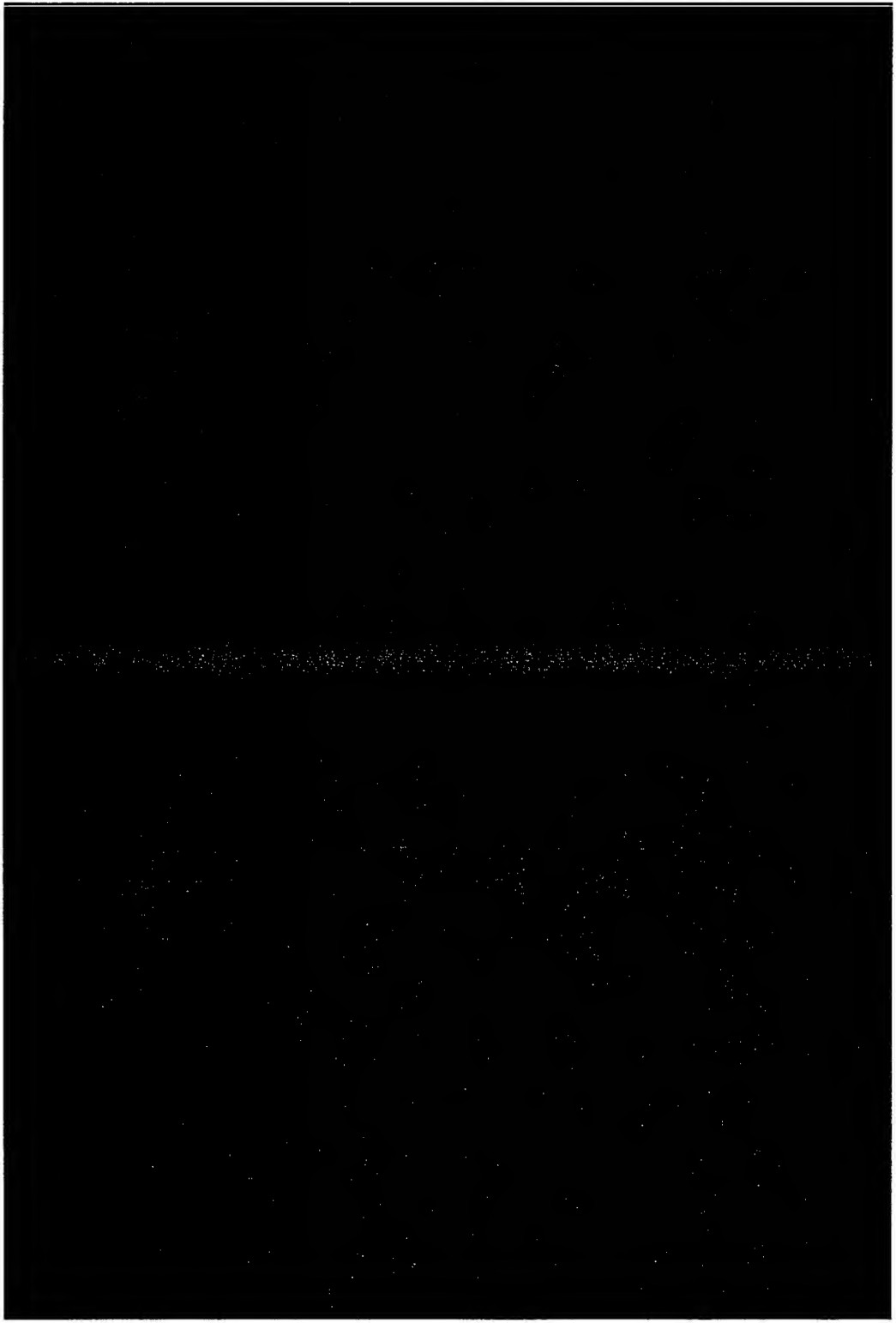


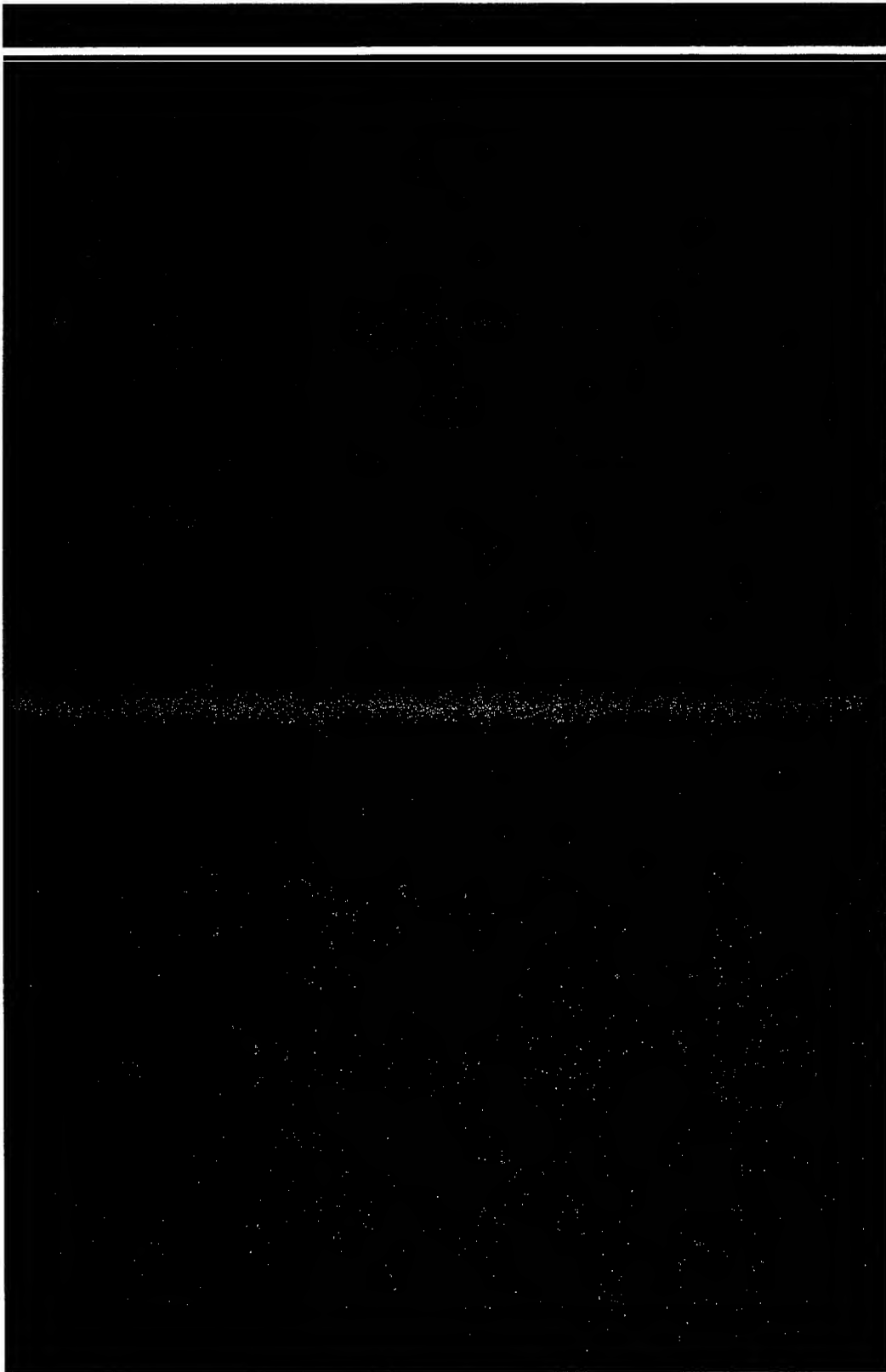
[REDACTED]

[REDACTED]

[REDACTED]









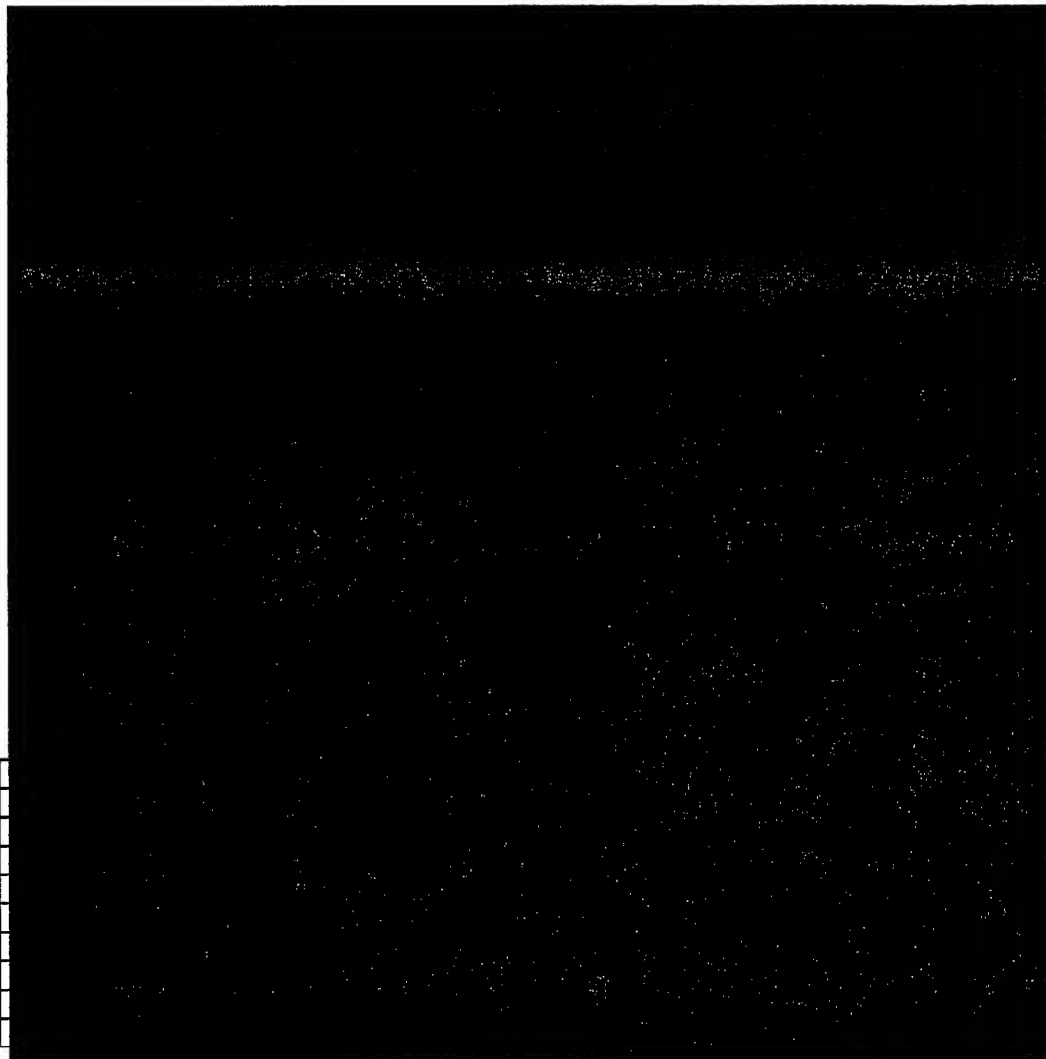
4 Interfaces and Schemas

4.1 Notification Submission

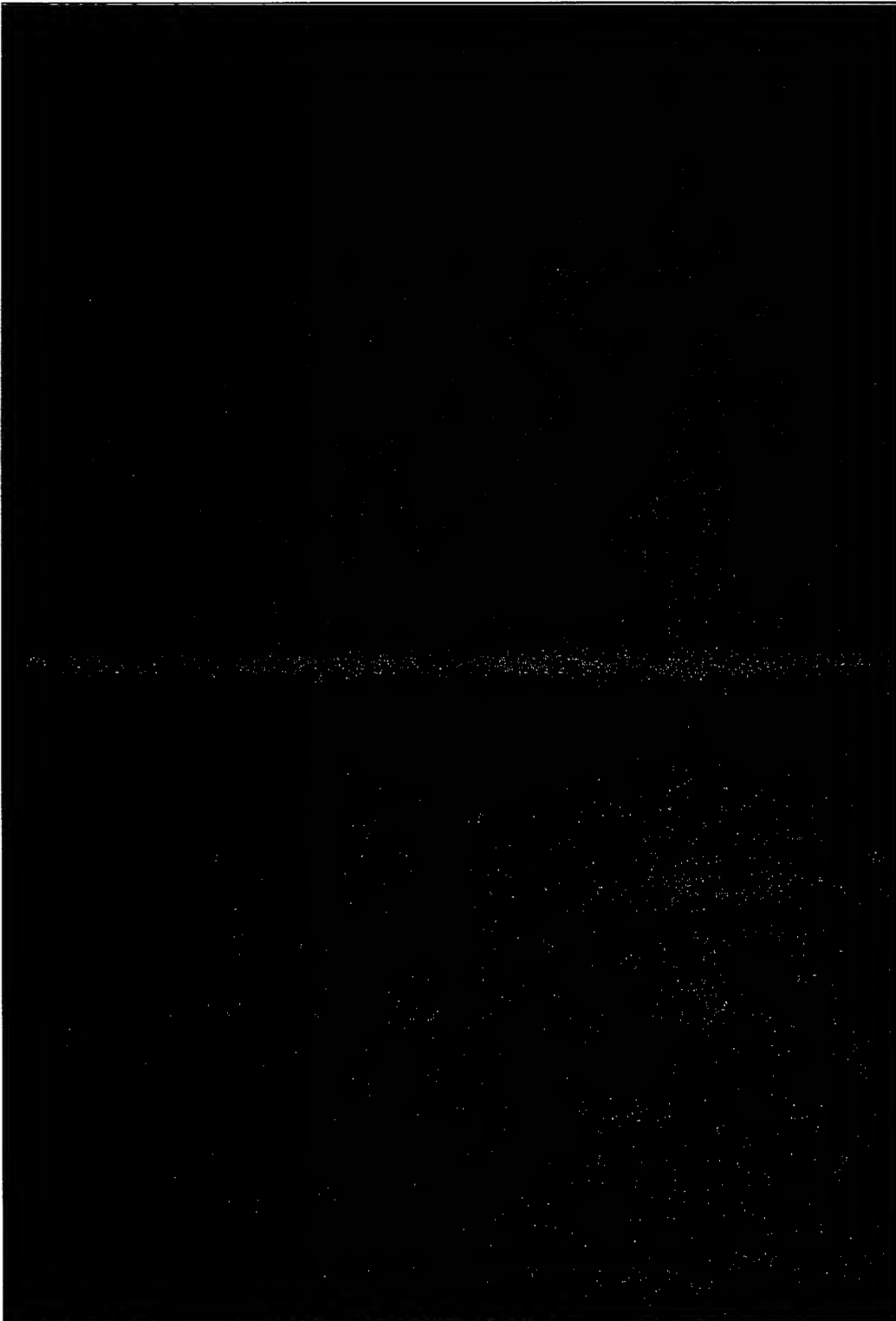
4.1.1 AUTD Notification Schema

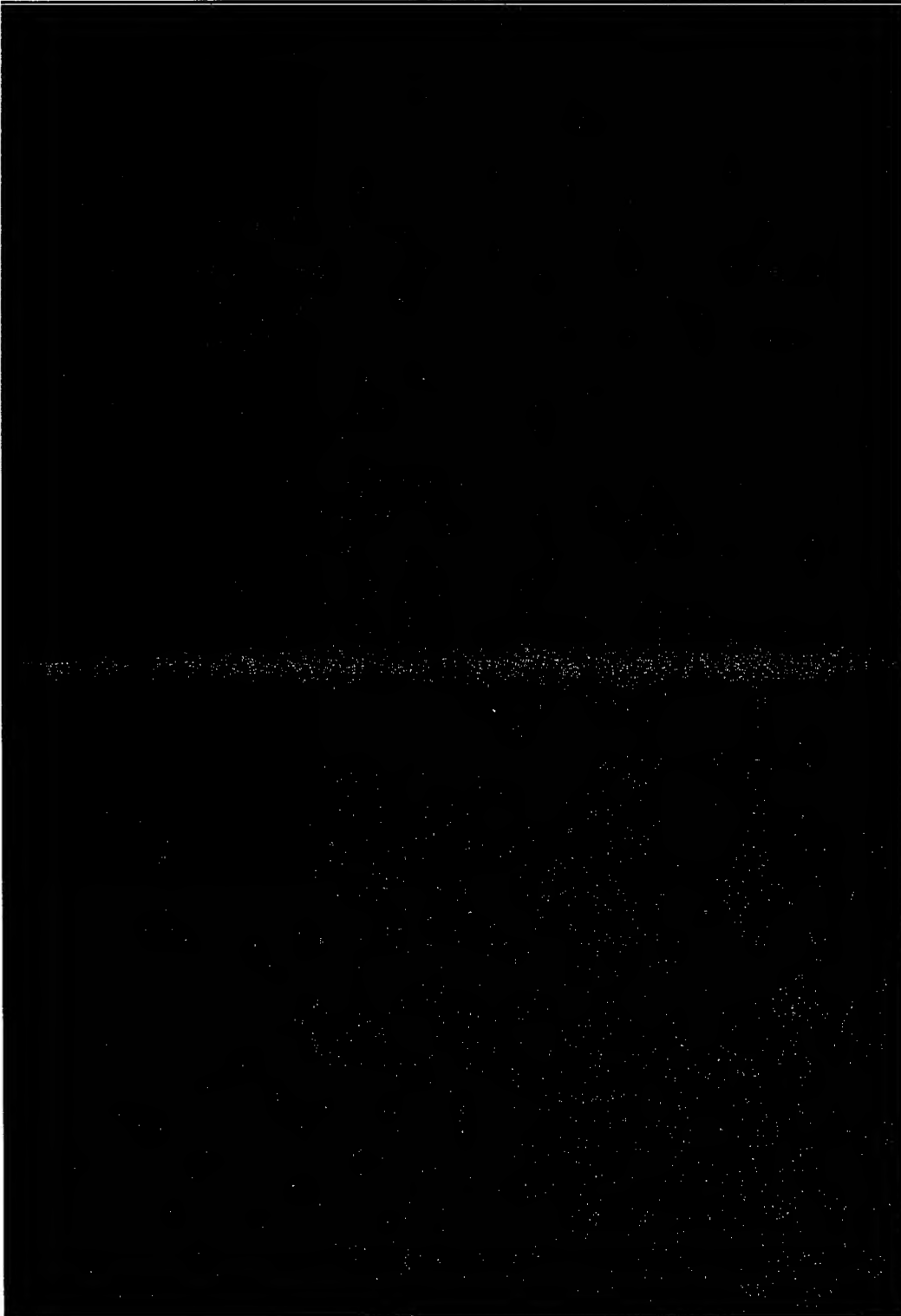
The notification will be sent to the AUTD device embedded in an SMS message. Since Nexus AUTD uses WBXml format which may contain embedded NULL chars in it, the complete binary stream will be Base64 encoded and then sent over SMS. The main content of the notification is the Sync GUID.

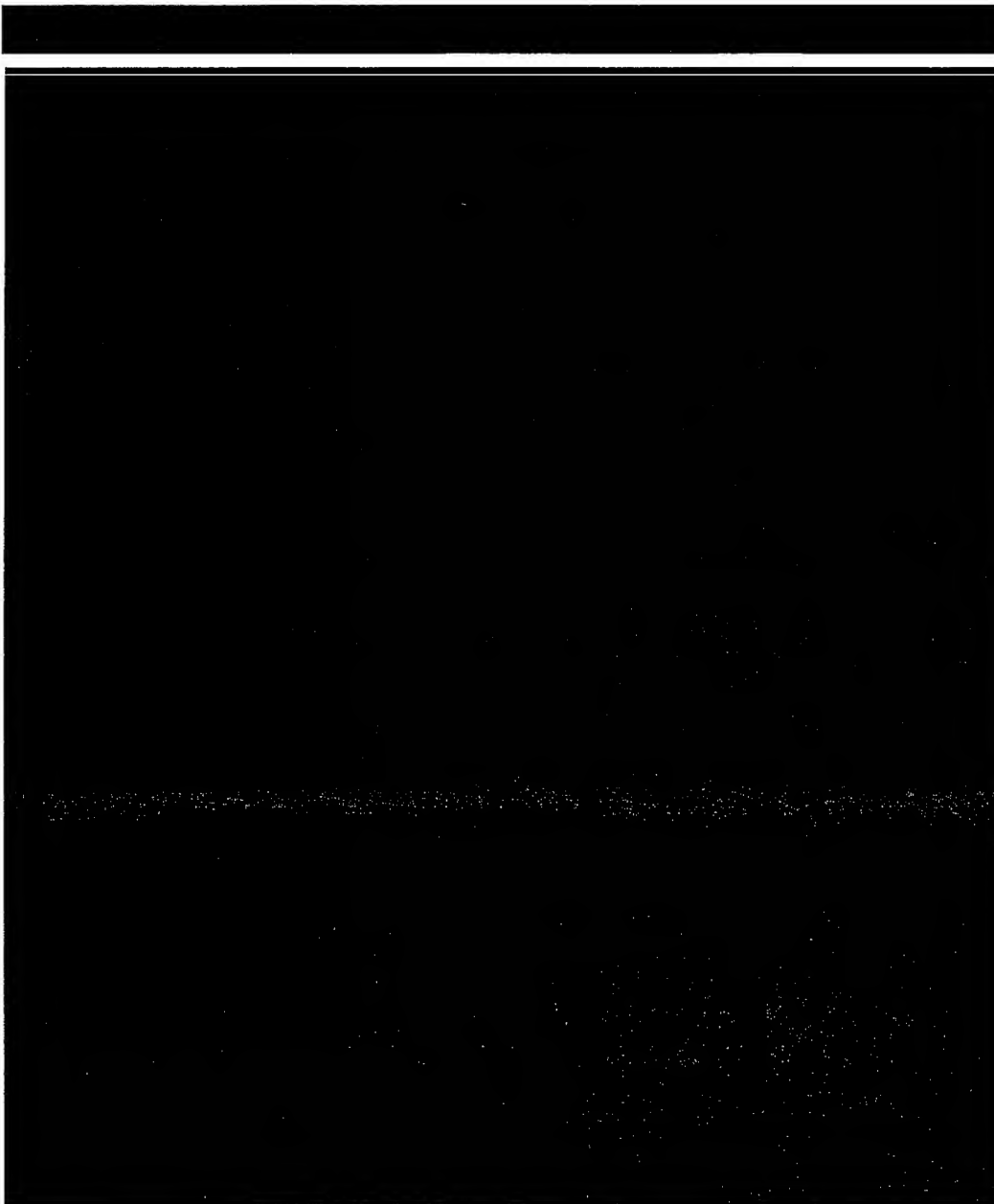
When the device receives the sync notification SMS message, it will get routed to Nexus sync handler by sending a WDP header specifying the destination port number. The header will conform to the standard WDP format specification as defined in the WDP spec: <http://www1.wapforum.org/tech/terms.asp?doc=WAP-259-WDP-20010614-a.pdf>











9 Glossary

Term	Definition
AUTD	Always-Up-To-Date: refers to the synchronization state between a mobile device and an Exchange mailbox
CLR	Common Language Runtime: The .Net run-time environment that manages the execution of code and provides services that make the development process easier.

Term	Definition
COM	Component Object Model: A platform-independent, distributed, object-oriented system for creating binary software components that can interact within or across process and machine boundaries.
DLL	Dynamic Link Library
EES	Exchange Event Source: the wireless component that will be installed on every Titanium Exchange server that hosts user mailboxes. The EES provides mobile notifications via store events.
IIS	Internet Information Services: A component of the Windows Server operating system under whose process the SMTP and HTTP components will run; the application name for this component and process is inetinfo.exe
In-Proc	In Process: Software that runs within the context of some other process rather than its own dedicated process.
MSMQ	Message Queuing: A technology that provides guaranteed message delivery, efficient routing, security, and priority-based messaging. This technology is supplied by the Windows operating system.
OMA	Outlook Mobility Access: A component of Titanium Exchange that will provide Sync, Browse, and Push functionality to mobile devices. The Push functionality is relevant to this design as it is responsible for delivering AUTD input to the MSN datacenter.
SMS	Short Messaging System: a text messaging protocol deployed by cellular phone carriers
SMTP	Simple Mail Transport Protocol: an industry standard protocol for sending email messages
Titanium OMA	The wireless component (was MIS) of the Exchange Titanium release
Titanium Exchange	The forthcoming release of Microsoft Exchange Server
TLS	Transport Layer Security: A negotiated encryption mechanism for transferring SMTP messages over an open network in a secure manner
WLBS	Windows Load Balancing Service
XML	Extensible Markup Language: An industry-standard language for data exchange between two disparate systems